

A Quasi-Random Approach to Matrix Spectral Analysis

Michael Ben-Or*

Lior Eldar[†]

April 7, 2017

Abstract

Inspired by the quantum computing algorithms for Linear Algebra problems [HHL09, TS13] we study how the simulation on a classical computer of this type of “Phase Estimation algorithms” performs when we apply it to solve the Eigen-Problem of Hermitian matrices. The result is a completely new, efficient and stable, parallel algorithm to compute an approximate spectral decomposition of any Hermitian matrix. The algorithm can be implemented by Boolean circuits in $O(\log^2 n)$ parallel time with a total cost of $O(n^{\omega+1})$ Boolean operations. This Boolean complexity matches the best known rigorous $O(\log^2 n)$ parallel time algorithms, but unlike those algorithms our algorithm is (logarithmically) stable, so further improvements may lead to practical implementations.

All previous efficient and rigorous approaches to solve the Eigen-Problem use randomization to avoid bad condition as we do too. Our algorithm makes further use of randomization in a completely new way, taking random powers of a unitary matrix to randomize the phases of its eigenvalues. Proving that a tiny Gaussian perturbation and a random polynomial power are sufficient to ensure almost pairwise independence of the phases (mod 2π) is the main technical contribution of this work. This randomization enables us, given a Hermitian matrix with well separated eigenvalues, to sample a random eigenvalue and produce an approximate eigenvector in $O(\log^2 n)$ parallel time and $O(n^\omega)$ Boolean complexity. We conjecture that further improvements of our method can provide a stable solution to the full approximate spectral decomposition problem with complexity similar to the complexity (up to a logarithmic factor) of sampling a single eigenvector.

1 Introduction

1.1 General

The eigen-problem of Hermitian matrices is the problem of computing the eigenvalues and eigenvectors of a Hermitian matrix. This problem is ubiquitous in computer science and engineering, and because of its relatively high computational complexity imposes a high computational load on most modern information processing systems.

The Abel-Ruffini theorem implies there is no deterministic expression for the eigenvalues of a general matrix A , and for this reason eigen-problem algorithms must be iterative. This gives rise to a host of problems: these algorithms are hard to analyze rigorously, and often turn out to be unstable, and hard to parallelize.

*School of Computer Science and Engineering, The Hebrew University, Jerusalem, Israel

[†]Center for Theoretical Physics, MIT, Email: leldar@mit.edu

As with many other problems in computer science, one typically considers an approximate spectral decomposition of a matrix. Thus, given a matrix A , we are usually interested not in its exact eigenvalues / eigenvectors, which may be very hard to compute, (and possibly very long to describe once computed), but rather in an approximate decomposition:

Definition 1. Approximate Spectral Decomposition - $\text{ASD}(A, \delta)$

Let A be some $n \times n$ Hermitian matrix. An approximate spectral decomposition of A , with accuracy parameter $\delta = 1/\text{poly}(n)$ is a set of unit vectors $\{v_i\}_{i=1}^n$, $\|v_i\| = 1$ such that each v_i has $\|Av_i - \lambda_i v_i\|_2 \leq \delta$, for some number $\lambda_i \in \mathbb{R}$, and

$$\left\| \sum_{i \in [n]} \lambda_i v_i v_i^T - A \right\| \leq \delta \|A\|,$$

where $\|X\|$ is the operator norm of X .

By standard arguments the above can be generalized to an arbitrary $n \times n$ matrix A , by considering the Hermitian matrix $A^H A$, in which case $\text{ASD}(A^H A, \delta)$ is an approximation of the singular vectors (and singular values) of A . We note that the definition of ASD then corresponds to a "smooth analysis" of matrices: namely given input A , we do not find a spectral decomposition of A , but rather the decomposition of a matrix A' , such that $\|A - A'\| \leq \delta$. We also point out, that the definition of ASD holds just as well in the case of nearly degenerate matrices: we do not require a one-to-one correspondence with the eigenvectors of A , which can be extremely hard to achieve, but rather to find some set of approximate eigenvectors, such that the corresponding weighted sum of rank-1 projections form an approximation of A .

When one considers an algorithm \mathcal{A} for the ASD problem, one can examine its *arithmetic* complexity or *boolean* complexity. The arithmetic complexity is the minimal size arithmetic circuit C (namely each node computes addition/multiplication/division to unbounded accuracy) that implements \mathcal{A} , whereas the boolean complexity counts the number of boolean AND/OR gates of fan-in 2 required to implement \mathcal{A} .

Given the definition above, and following Demmel et al. [DDH07] we consider an algorithm \mathcal{A} to be log-stable (or stable for short), if there exists a circuit C that implements \mathcal{A} on $n \times n$ matrices, and a number $t = O(\log(n))$, such that each arithmetic computation in C uses at most t bits of precision, and output of the circuit deviates from the output of the arithmetic circuit by at most $1/\text{poly}(n)$. We note that when an algorithm is *stable* then its boolean complexity is equal to its arithmetic complexity up to a factor $O(\log(n))$. If, however, an algorithm is *unstable* then its boolean complexity could be larger by a factor of up to n . In the study of practical numerical linear algebra algorithms, one usually identifies algorithms that are stable with "practical", and algorithms that are not stable to be impractical. This usually, because the computing machines are restricted to representing numbers with a number of bits that is a small fraction of the size of the input.

In terms of parallelism, we will refer to the complexity class $\text{NC}^{(k)}$ (see Definition 10) which is the set of all computational problems that can be solved by uniform Boolean circuits of size $\text{poly}(n)$ in time $O(\log^k(n))$. Often, we will refer to the class $\text{RNC}^{(k)}$, in which the parallel $\text{NC}^{(k)}$ circuit is also allowed to accept uniform random bits. One would like an ASD algorithm to have minimal arithmetic / boolean complexity, and minimal parallel time. Ideally, one would also like this algorithm to be stable.

1.2 Main Contribution

Inspired by recent quantum computing algorithms [HHL09, TS13], we introduce a new perspective on the problem of computing the ASD that is based on low-discrepancy sequences. Roughly speaking, low-discrepancy sequences are deterministic sequences which appear to be

random, because they “visit” each small sub-cube the same number of times that a completely random sequence would up to a small additive error.

Definition 2. Multi-dimensional Discrepancy

For integer s , put $I^s = [0, 1]^s$. Given a sequence $x = (x_n)_{n=1}^N$, with $x_n \in I^s$ the discrepancy $D_M(x)$ is defined as:

$$D_M(x) = \sup_{B \in \mathcal{B}} \left\{ \left| \frac{1}{M} \sum_{n=1}^M \chi_B(x_n) - \text{vol}(B) \right| \right\},$$

where $\chi_B(x_n)$ is an indicator function which is 1 if $x_n \in B$ is the set of all s -products of intervals $\prod_{i=1}^s [u_i, v_i]$, with $[u_i, v_i] \pmod{1} \subseteq [0, 1]$.

We recast the ASD problem as a question about the discrepancy of a certain sequence related to the input matrix. Specifically, given a Hermitian matrix A with, say n unique eigenvalues $\{\lambda_i\}_{i \in [n]}$ the central object of interest is the sequence comprised of n -dimensional vectors of eigenvalue residuals:

$$S(A) = (\{\lambda_1 \cdot 1\}, \dots, \{\lambda_n \cdot 1\}), (\{\lambda_1 \cdot 2\}, \dots, \{\lambda_n \cdot 2\}), \dots, (\{\lambda_1 \cdot M\}, \dots, \{\lambda_n \cdot M\}),$$

where $\{x\}$ is the fractional part of $x \in \mathbb{R}$, and $M = \text{poly}(n)$ is some large integer. $S(A)$ is hence a sequence of length M in $[0, 1]^n$. We would like $S(A)$ to have as small discrepancy as possible. Hence, in sharp contrast to previous algorithms, instead of the computational effort being concentrated on revealing “structure” in the matrix, our algorithm is actually focused on producing random-behaving dynamics.

The main application of our approach presented in this paper is a new stable and parallel and stable algorithm for computing the ASD of any Hermitian matrix.

Theorem 3. For any Hermitian matrix $0 \preceq A \preceq 0.9I$, and $\delta \leq n^{-7}$ we have $\text{ASD}(A, \delta) \in \text{RNC}^{(2)}$, with total boolean complexity $O(n^{\omega+1})$. The algorithm is log-stable.

The boolean complexity of our algorithm is $O(n^{\omega+1})$. If however, one is interested in sampling a uniformly random eigenvector, it can be achieved in complexity $O(n^\omega)$.¹

1.3 Overview of the Algorithm

To compute the ASD of a given matrix A , we first consider a similar problem of sampling uniformly an approximate eigenvector of A , where the eigenvalues of A are assumed to be well-separated. Clearly, if one can sample from this distribution in RNC^2 , then by the coupon collector’s bound concatenating $O(n \log(n))$ many parallel copies of this routine, one can sample all eigenvectors quickly with high probability. To do this, we require a definition of a Hermitian matrix that is δ -separated:

Definition 4. δ -separated

Let A be an $n \times n$ PSD matrix. A has a complete set of real eigenvalues $\lambda_1 > \lambda_2 > \dots > \lambda_n \geq 0$. We say that A is δ -separated if $\lambda_j - \lambda_{j+1} \geq \delta$ for all $j < n$, and $\lambda_1 \leq 1 - \delta$.

Next, we introduce the notion of a separating integer w.r.t. a sequence of real numbers:

Definition 5. Separating Integer

Let $\bar{\lambda} = (\lambda_1, \dots, \lambda_n) \in [0, 1]^n$. For $\alpha > 4$ define

$$B_{\text{out}} = [1 - 1/(4n), -1 + 1/(4n)] \quad \text{and} \quad B_{\text{in}}(\alpha) = [-1/(\alpha n), 1/(\alpha n)],$$

A positive integer m is said to separate the k -th element of $\bar{\lambda}$ w.r.t. $B_{\text{in}}, B_{\text{out}}$ if it satisfies:

¹ ω signifies the infimum over all constants c such that one can multiply two matrices in time at most n^c .

- $\{m\lambda_k\} \in B_{in}(\alpha)$
- $\forall j \neq k \ \{m\lambda_j\} \notin B_{out}$

and finally define the notion of a separating integer w.r.t. a δ -separated matrix.

Definition 6. A positive integer m is said to separate k in a δ -separated matrix A w.r.t. B_{in}, B_{out} , if m separates the k -th element of $\mathcal{L}(A)$ w.r.t. B_{in}, B_{out} .

Following is a sketch of the main sampling routine. For complete details see Section 5. The routine accepts a separating integer m of the i -th eigenvalue of a δ -separated matrix A , a precision parameter δ and returns a δ approximation of the i -th eigenvector of A :

Algorithm 7. $\text{Filter}(A, m, \delta)$

1. **Compute parameters:**

$$p = 24n^2 \lceil \ln(1/\delta) \rceil, \zeta = \delta^2/(2pm).$$

2. **Sample random unit vector:**

Sample a standard complex Gaussian vector v , set $w_0 = v/\|v\|$.

3. **Approximate matrix exponent:**

Compute a ζ Taylor approximation of e^{iA} , denoted by \tilde{U} .

4. **Raise to power:**

Compute \tilde{U}^m by repeated squaring.

5. **Generate matrix polynomial:**

Compute $B = \left(\frac{I + \tilde{U}^m}{2} \right)^p$ by repeated squaring.

6. **Filter:**

Compute $w = \frac{B \cdot w_0}{\|B \cdot w_0\|}$.

7. **Decide:**

Set $z = A \cdot w$, $i_0 = \arg \max_{i \in [n]} |w_i|$ and compute $c = z_{i_0}/w_{i_0}$. If

$$\|A \cdot w - c \cdot w\| \leq 3\delta\sqrt{n}$$

return w , and otherwise reject.

In words - the algorithm samples a random vector and then multiplies it essentially by the matrix $B = ((I + e^{iAm})/2)^p$. After this "filtering" step, it evaluates whether or not the resulting vector is close to being an eigenvector of A , and keep it if it is. To understand the behavior of the algorithm, it is insightful to consider the behavior in the eigenbasis of A .

$$w = \sum_i \alpha_i w_i,$$

where $\{w_i\}_{i \in [n]}$ is an orthonormal basis for A corresponding to eigenvalues $\{\lambda_i\}_{i \in [n]}$. If $\{m\lambda_i\}$, i.e. - the fractional part of $m\lambda_i$, is very close to 0, and $\{m\lambda_j\}$ is $\sim 2 \ln n/p$ far from 0 for all $j \neq i$, then after multiplication by B and normalization, all eigenvectors w_j for $j \neq i$ are attenuated by factor $1/n^2$ relative to w_i , and hence the resulting vector is $1/n$ close to an eigenvector of λ_i .

Hence, a sufficient condition on the number m that would imply that $w = \text{Filter}(A, m, \delta)$ is an approximation of the i -th eigenvector is the following property: $\{m\lambda_i\}$ is very close to 0,

and for all $j \neq i$ $\{m\lambda_j\}$ is bounded away from 0. This corresponds to the fact that m separates i in A , as assumed.

So to sample uniformly an approximate eigenvector, we would like to call $\text{Filter}(A, m, \delta)$ for $m \sim U[M]$ for $M = \text{poly}(n)$ such that m separates i where $i \sim U[n]$. The main observation here, is that this condition is satisfied if the sequence of residuals of integer multiples of the eigenvalues $S(A)$ defined above has the aforementioned *low discrepancy* property.

Most of the work in this study is devoted to achieving this property. Computationally, we achieve low-discrepancy of $S(A)$ simply by additive Gaussian perturbation prior to calling the sampling routine. We show that if we perturb a matrix using a Gaussian matrix \mathcal{E} of variance $1/\text{poly}(n)$, then $S(A + \mathcal{E})$ has discrepancy which is $1/\text{poly}(n)$. Showing this is non-trivial because arbitrary vectors of eigenvalues $\lambda_1, \dots, \lambda_n$ do not generate low-discrepancy sequences in general ², and on the other hand we are also severely limited in our ability to perturb the eigenvalues without deviating too much from the original matrix. This is the subject of our main technical theorem 32, which may be of independent interest:

Theorem. (sketch) Let A be an $n \times n$ Hermitian matrix, and \mathcal{E} be a standard Gaussian matrix. For any $a > 0, b > 0$ there exists $M = M(a, b) = \text{poly}(n)$ such that w.p. at least $1 - n^{-b}$ the sequence of residuals of eigenvalue multiples of $A + n^{-a} \cdot \mathcal{E}$ of length M has discrepancy at most n^{-b} .

Perturbing the input matrix has the additional benefit of making sure that A has a exactly n unique eigenvalues with high probability. This follows from a breakthrough theorem by Nguyen, Tao and Vu [NTV16] which has provided a resolution of this long-standing open problem, which was considered unproven folklore until that point. This theorem allows us to handle general Hermitian matrices without extra conditions on the conditioning number of A / its eigenvalue spacing.

1.4 Prior Art

There are numerous algorithms for computing the ASD of a matrix, relying most prominently on the QR decomposition [TB97]. For specific types of matrices, like tridiagonal matrices much faster algorithms are known [Rei05], but here we consider the most general Hermitian case. We summarize the state of the art algorithms for this problems in terms of their complexity (boolean / arithmetic, serial / parallel) and compare them to our own:

	Arithmetic Complexity	Boolean Complexity	Parallel Time	Log-Stable	Comments
Csanky	$\tilde{O}(n^{\omega+1})$	$\tilde{O}(n^{\omega+2})$	$\log^2(n)$	NO	
Demmel et al. [DDH07]	$\tilde{O}(n^\omega)$	$\tilde{O}(n^\omega)(*)$	N/A	YES	* Conjectured by us, by modifying the algorithm.
Bini et al., Reif [BP92, Rei05]	$\tilde{O}(n^\omega)$	$\tilde{O}(n^{\omega+1})$	$O(\log^2(n))$	NO	Working with $\Omega(n)$ bit Integers
New	$\tilde{O}(n^{\omega+1})$	$\tilde{O}(n^{\omega+1})$	$\log^2(n)$	YES	

Comparing our algorithm to the best known $\text{NC}^{(2)}$ algorithms, it is more efficient by a factor of n compared with Csanky's algorithm [Koz92]. Notably, our algorithm is completely disjoint

²Consider for example the sequence of values $1, 1/2, \dots, 1/n$. Then multiplying these numbers individually by a random integer m and taking the residual would map to 0 all values $1/i$ for which $i|m$, and there is a logarithmic number of these on average. This sequence of eigenvalues is well-separated, and at least potentially could arise from a random matrix. We show, however, that this is not the typical case.

from Csanky’s techniques - which rely on computing explicitly high powers of the input matrix, and computes the characteristic polynomial of the matrix using the Newton identities on the traces of those powers. This is an inherently unstable algorithm as it finds the eigenvalues by approximating the roots of the characteristic polynomial and small perturbation to the coefficients of the polynomial may lead to large deviations of the roots.

The algorithms of Demmel et al., Bini et al. and Reif, rely on efficient implementation of variants of the QR algorithm. Our asymptotic bounds are worse than Demmel et al. in terms of total arithmetic/boolean complexity, though we conjecture that this is an artifact of our proof strategy, and not an inherent problem (see the section on open problems), and in fact, a variant of the algorithm could probably achieve a boolean complexity of $O(n^\omega)$. We note that the QR algorithm is not known to be parallelizable in a stable way, and hence the fast parallel algorithms of Bini et al. and Reif are not stable and probably impractical. In fact the QR decomposition has been shown, for standard implementations like the Given’s or Householder method, to be P -complete [LMM99] assuming the real-RAM model. Thus, it is unlikely to be stably-parallelizable unless $P = NC$.³

Thus, to the best of our knowledge, our algorithm is the first parallel algorithm for the ASD of general Hermitian matrices that is both parallel and stable. In particular it achieves the smallest bit-complexity of any $RNC^{(2)}$ algorithm to date. We conjecture that our approach may present a practical and parallel alternative to computing the ASD. We dwell on this point a bit more in Section 8.

1.4.1 Comparison to the power method / QR algorithm

An arguably natural benchmark by which to test the novelty of the proposed algorithm is the iterative power-method for computing the eigenvalues of a Hermitian matrix. In this method, one starts from some random vector b_0 , and at each iteration k sets:

$$b_{k+1} = \frac{Ab_k}{\|Ab_k\|}.$$

This method can achieve polynomially good approximation of the top eigenvalue in time which is logarithmic in n , for A with a constant spectral-gap.

Both the power method and our proposed scheme are similar in the sense that they attempt to extract the eigenvectors of the input matrix directly. Also, if two eigenvalues are ε -close in magnitude, for some $\varepsilon > 0$, then they require essentially the same exponent of A in the power method, and of $e^{2\pi i A}$ in our scheme to distinguish between them. However, the similarity stops here. We maintain, that the power method is both conceptually different, and for general Hermitian matrices performs much worse, in terms of running time, compared with our proposed algorithm.

Conceptually, in the power method, we seek to leverage the difference in *magnitude* between adjacent eigenvalues in order to extract the eigenvectors. On the other hand, in our proposed scheme we recast the problem on the unit sphere $S^{(1)}$, where we are interested in the spacing of the residuals of integer multiples of the eigenvalues. Worded differently, our setting exploits the additive group structure of the eigenvalues modulo 1, whereas the power method distinguishes between them multiplicatively. In the additive group setting, the advantage is that we can consider the discrepancy of the sequence of residuals, and analyze how quickly these residuals mimic a completely independent random distribution. Furthermore, in the additive setting there is inherent symmetry between the eigenvalues, as no eigenvalue is more likely to

³We point out that the algorithm of Reif [Rei05] achieves a QR factorization in parallel time $O(\log^2(n))$ in the arithmetic model, thus showing that QR is indeed parallelizable, but it relies on computations modulo large integers and therefore not stable and not practical.

be sampled than another. This allows for a natural parallelization of the algorithm to extract simultaneously approximation of all eigenvectors.

The well-known QR algorithm for eigendecomposition [GVL96] is the de-facto standard for computing the ASD, and is, in a sense, a parallel version of the power-method. That algorithm applies an iterated sequence of QR decompositions: At each step k we compute (where $A_1 = A$ - the input matrix)

$$A_k = Q_k R_k,$$

and then set

$$A_{k+1} = R_k Q_k.$$

The algorithm runs in time $\tilde{O}(n^3)$, by applying several pre-processing steps [GVL96], and the fast variant of Demmel et al. in time $O(n^\omega)$. However, as stated above, the QR decomposition which is at the core of these methods is not known to be stably parallel.

1.5 Open Questions

We outline several open questions that may be interesting to research following this work:

1. Is it possible to attain a serial run-time of $O(n^\omega)$ for this algorithm? We conjecture that this is possible based on numerical evidence for a variant of this algorithm, yet we do not have a proof of this fact. While not directly improving on the best previously known serial run-time ([DDH07]), it would reduce the overall work performed by our parallel $\text{RNC}^{(2)}$ algorithm to match the work done by state-of-the-art serial run-time algorithm.
2. What other linear-algebra algorithms can be designed using our methods? We would like these algorithms to improve on previous algorithms in either the stability, boolean complexity, parallel run-time, or all these parameters together.
3. Could one reduce the number of random bits required by the algorithm? Currently - we show that using $\tilde{O}(n^2)$ random bits - i.e. applying additive Gaussian perturbation results in a matrix whose eigenvalues seed a low-discrepancy sequence. However, can one do away with only $\tilde{O}(n)$ random bits - by applying a tri-diagonal perturbation to the matrix?
4. Is our algorithm practical? What is the actual run-time of the algorithm on matrices of "reasonable" size, and does it compare with state-of-the art parallel algorithms? Our numerical evidence suggests that in practice our algorithm may run much faster than the analytical asymptotic bounds we provide here.

Table of contents

2	Preliminaries	9
3	Additive Perturbation	10
4	Low-Discrepancy Sequences	12
5	A Filtering Algorithm	16
6	Sampling Separating Integers	20
7	Parallel Algorithm for ASD	26
8	Numerical Experiments	30

2 Preliminaries

2.1 Notation

A random variable x distributed according to distribution \mathcal{D} is denoted by $x \sim \mathcal{D}$. For a matrix X , $\|X\|$ signifies the operator norm of X . For a set S , $U[S]$ is the uniform distribution on S . For integer $M > 0$ the set $[M]$ is the set of integers $\{0, 1, \dots, M-1\}$. For real number x , $\{x\}$ denotes the fractional part of x : $\{x\} = x - \lfloor x \rfloor$. For a Hermitian $n \times n$ matrix A , with eigenvalues $\{\lambda_i\}_{i=1}^n$, $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ $\mathcal{L}(A) = (\lambda_1, \dots, \lambda_n) \in \mathbb{R}^n$ denotes the vector of sorted eigenvalues of A . For a measurable subset $S \subseteq \mathbb{R}^n$ $\text{vol}(S)$ denotes the volume of S . Φ is the empty set. GUE is the global unitary ensemble of random matrices. $\mathbb{N}, \mathbb{Z}, \mathbb{C}$ signify the natural, integer, and complex numbers, respectively. For a matrix A , A^H is the Hermitian conjugate-transpose of A . For number $n > 0$ $\ln n$ denotes the natural logarithm, and $\log n$ denotes the binary logarithm. $\mu(\eta, \sigma^2)$ is the standard Gaussian measure with mean η and variance σ^2 . $U(n)$ is the set of $n \times n$ unitary matrices.

2.2 Definitions

2.2.1 Hermitian matrices

We repeat again the definition of the ASD due to its importance:

Definition 8. Approximate Spectral Decomposition - ASD(A, δ)

Let A be some $n \times n$ δ -separated Hermitian matrix. An approximate spectral decomposition of A , with accuracy parameter $\delta = 1/\text{poly}(n)$ is a set of unit vectors $\{v_i\}_{i=1}^n$, $\|v_i\| = 1$ such that each v_i has $\|Av_i - \lambda_i v_i\|_2 \leq \delta$, for some number $\lambda_i \in \mathbb{C}$, and

$$\left\| \sum_{i \in [n]} \lambda_i v_i v_i^T - A \right\| \leq \delta,$$

where $\|X\|$ is the operator norm of X .

By standard arguments the above can be generalized to an arbitrary $n \times n$ matrix A , by considering the Hermitian matrix $A^H A$, in which case $\text{ASD}(A^H A, \delta)$ is an approximation of the singular vectors (and singular values) of A . We note that if $\{v_i\}_{i \in [n]}$ is an $\text{ASD}(A, \delta)$ for some matrix A , and B is a matrix such that $\|A - B\| \leq \delta$, then by the triangle inequality $\{v_i\}_{i \in [n]}$ is an $\text{ASD}(B, \sqrt{2}\delta)$.

2.2.2 Complexity

Definition 9. Let ω denote the infimum over all t such that any two $n \times n$ matrices can be multiplied using a number of products at most n^t .

The current best upper-bound on ω is 2.372 due to Williams [Wil12].

Definition 10. Class NC

The class $\text{NC}^{(k)}$ is the set of problems computed by uniform boolean circuits, with a polynomial number of gates, and depth at most $O(\log^k n)$.

We will require the following known fact:

Fact 11. [Ak190] There exists an algorithm for sorting n numbers in time $\log(n)$, using n processors.

Definition 12. Class RNC

The class $\text{RNC}^{(k)}$ is the set of problems that can be computed by uniform boolean circuits, with a polynomial number of gates, accepting a polynomial number of random bits, and depth at most $O(\log^k n)$.

For simplicity, we shall assume in this work that RNC circuits are allowed to accept t -bit numbers, sampled from a truncated Gaussian distribution, and discretized to t -bits of precision.

2.2.3 Stable Computation

Following Demmel et al. [DDH07] we define the notion of log-stability as one where truncating each binary arithmetic operation to $O(\log(n))$ bits of precision doesn't change the result by much:

Definition 13. *(t, δ) -stable randomized computation*

Let C denote a randomized arithmetic circuit, and \mathcal{D} be its output distribution supported on \mathbb{R}^n . Let D denote the discretization of C to t bits as follows: each infinite-precision arithmetic operation is followed by rounding to t bits. Let \mathcal{D}' denote the output distribution of D . C is said to be (t, δ) -stable if

$$\forall x \exists y, \mathcal{D}(x) = \mathcal{D}'(y) \text{ and } \|x - y\| \leq \delta.$$

Definition 14. *Log-stable computation*

Let C be a randomized arithmetic circuit that accepts n input numbers. C is said to be log-stable if for any $\delta = 1/\text{poly}(n)$ it is (t, δ) -stable for some $t = O(\log(1/\delta))$.

3 Additive Perturbation

Matrix perturbation is a well-developed theory [SS90, GVL96] examining the behavior of eigenvalues and eigen-vectors under additive perturbation, usually much smaller compared to the norm of the original matrix. While general eigenvalue problems are usually unstable against perturbation, for Hermitian matrices the situation is much better: the Bauer-Fike theorem states that the perturbed eigenvalues can only deviate from the original eigenvalues by an amount corresponding to the relative strength of the perturbation.

In particular, when the perturbed matrix A is δ -separated one can compute an explicit estimate for the behavior of the perturbed eigenvalues. We use here a quantitative estimate by [SS90]:

Fact 15. *Rayleigh quotient for well-separated eigenvalues*

Let A be a δ -separated $n \times n$ Hermitian matrix with eigenvalues $\lambda_1 > \lambda_2 > \dots > \lambda_n$, and corresponding orthonormal basis $\{v_i\}_{i \in [n]}$. Let \mathcal{E} be an additive perturbation of A satisfying $|\mathcal{E}_{i,j}| \leq \varepsilon$ for all i, j . Let $\tilde{\lambda}_i$ denote the i -th eigenvalue of $A + \mathcal{E}$. There exists a constant $c > 0$ satisfying:

$$\forall i \in [n] \quad \tilde{\lambda}_i = \lambda_i + v_i^H \mathcal{E} v_i + \zeta_i, \quad |\zeta_i| \leq c\varepsilon^2/\delta.$$

In fact, if the perturbation \mathcal{E} is GUE a stronger characterization is readily available:

Corollary 16. *Let A be a δ -separated $n \times n$ Hermitian matrix with eigenvalues $\{\lambda_i\}_{i \in [n]}$, and corresponding orthonormal basis $\{v_i\}_{i \in [n]}$. Let \mathcal{E} be GUE. Then the eigenvalues $\{\lambda'_i\}_{i \in [n]}$ of the perturbed matrix $A' = A + \varepsilon \cdot \mathcal{E}$ are distributed as follows:*

$$\forall i \in [n] \quad \tilde{\lambda}_i = (1 - \alpha) \cdot \mu(\lambda_i, \varepsilon^2) + \alpha \cdot \mathcal{D} + \zeta_i, \quad 0 \leq \alpha \leq 2^{-n}, |\zeta_i| \leq 16cn \cdot \varepsilon^2/\delta$$

for some distribution \mathcal{D} .

Proof. By Fact 15 the eigenvalues λ'_i behave as

$$\lambda'_i = \lambda_i + v_i^H \mathcal{E} v_i + \zeta_i, \quad |\zeta_i| \leq c \max_{i,j} |\mathcal{E}_{i,j}|^2/\delta.$$

The standard Gaussian satisfies:

$$P_\mu(|x| \geq 4\sqrt{n}) \leq 2^{-2n}.$$

Thus, by the union bound we have that $|\mathcal{E}_{i,j}| \leq 4\sqrt{n}$ for all i, j w.p. at least $1 - 2^{-n}$. Hence, w.p. at least $1 - 2^{-n}$ we have:

$$\forall i \in [n] \quad |\zeta_i| \leq c \max_{i,j} |\mathcal{E}_{i,j}|^2 / \delta \leq 16cn \cdot \varepsilon^2 / \delta,$$

Suppose that this is the case, and let \mathcal{E}' denote the GUE matrix, *conditioned* on having bounded entries. We can write:

$$\mathcal{E}' = (1 - \alpha) \cdot \mathcal{E} + \alpha \cdot \mathcal{D},$$

where \mathcal{D} is some distribution on $n \times n$ matrices and $0 \leq \alpha \leq 2^{-n}$. The distribution \mathcal{E} is invariant under unitary conjugation, i.e.:

$$\forall u \in U(n) \quad u\mathcal{E}u^H = \mathcal{E}$$

then

$$\forall u \in U(n) \quad u\mathcal{E}'u^H = (1 - \alpha) \cdot \mathcal{E} + \alpha \cdot \mathcal{D}',$$

where $\mathcal{D}' = u\mathcal{D}u^H$ is some distribution on $n \times n$ matrices. Hence, up to statistical distance at most α we can assume w.l.o.g. that $v_i = e_i$. This implies that

$$\lambda'_i = \lambda_i + (1 - \alpha) \cdot \mathcal{E}_{i,i} + \alpha \cdot \mathcal{D}' + \zeta_i,$$

where $|\zeta_i| \leq 16cn \cdot \varepsilon^2 / \delta$. Since $\mathcal{E}_{i,i} = \mu(0, \varepsilon^2)$ then

$$\lambda'_i = (1 - \alpha) \cdot \mu(\lambda_i, \varepsilon^2) + \alpha \cdot \mathcal{D}' + \zeta_i, \quad |\zeta_i| \leq 16cn \cdot \varepsilon^2 / \delta.$$

□

The stability of eigenvalues has also been generalized to eigenvectors and even general invariant subspaces [SS90] in the following sense: if there is a cluster of eigenvalues that is “well-separated” from all other eigenvalues, then the orthogonal projection onto the subspace spanned by the corresponding eigenvectors is likewise stable under additive perturbation whose scale is negligible compared to the separation of the eigenvalue of this cluster from the rest of the spectrum. In particular, if a matrix is δ -separated, then its eigenvectors are individually stable as follows:

Fact 17. [SS90] *Rayleigh quotient for well-separated invariant subspaces*

Let A be a δ -separated $n \times n$ Hermitian matrix with eigenbasis $\{v_i\}_{i \in [n]}$, and

$$A_1 = A + \mathcal{E}, \quad \|\mathcal{E}\| \leq \varepsilon.$$

There exists an orthonormal basis $\{v'_i\}_{i \in [n]}$ and a constant $c > 0$ satisfying

$$\forall i \in [n] \quad \|v_i - v'_i\| \leq c\varepsilon^2 / \delta.$$

Our interest in additive perturbation, however, is not confined just to “stability” arguments. In fact, our main reason for using perturbation is to cause a scattering of the eigenvalues. The first step of our algorithm in fact applies additive perturbation to provide a minimal spacing between eigenvalues. Recently Nguyen et al. [NTV16] have provided the first proof that applying additive perturbation to any Hermitian matrix using a so-called Wigner ensemble, an ensemble of random matrices that generalize GUE, in fact causes the eigenvalues of the perturbed matrix to achieve a minimal inverse polynomial separation. We state their result:

Lemma 18. [NTV16] *Minimal eigenvalue spacing*

Let $M_n = F_n + \varepsilon \cdot X_n$, where F_n is a real symmetric matrix, $\|F_n\|_2 \leq 1$, $\varepsilon = n^{-\gamma}$ for some constant $\gamma > 0$, and X_n is GUE. Let $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ denote the eigenvalues of M_n , and put $\alpha_i = \lambda_i - \lambda_{i+1}$ for all $i < n$. Then for any fixed $A > 0$ there exists $B = B(\varepsilon) > 0$, such that

$$\max_{i \in [n]} \mathbb{P}(\alpha_i \leq n^{-B}) = O(n^{-A}).$$

In particular ⁴ for any $A > 0$ there exists $B > 0$ such that

$$\mathbb{P}\left(\min_{i \in [n]} \alpha_i \geq n^{-B}\right) = 1 - O(n^{-A}).$$

Using the lemma above we define:

Definition 19. For any $\delta = 1/\text{poly}(n)$, let $B^*(\delta)$ denote the largest number $B > 0$ such that for every F_n the matrix $M_n = F_n + \delta X_n$ satisfies:

$$\mathbb{P}\left(\min_{i \in [n]} \alpha_i \geq n^{-B}\right) \geq 0.99$$

4 Low-Discrepancy Sequences

4.1 Basic Introduction

Low discrepancy sequences (or “quasi-random” sequences) are a powerful tool in random sampling methods. Roughly speaking, these are deterministic sequences that visit any “reasonable” subset B a number of times that is roughly proportional to the volume of B , up to some small additive error, called the discrepancy.

Definition 20. Multi-dimensional discrepancy

For integer s , put $I^s = [0, 1]^s$. Given a sequence $x = (x_n)_{n=1}^N$, with $x_n \in I^s$ the discrepancy $D_N(x)$ is defined as:

$$D_N(x) = \sup_{B \in \mathcal{B}} \left\{ \left| \frac{1}{N} \sum_{n=1}^N \chi_B(x_n) - \text{vol}(B) \right| \right\},$$

where $\chi_B(x_n)$ is an indicator function which is 1 if $x_n \in B$ and 0 otherwise, and \mathcal{B} is a non-empty family of Lebesgue-measurable subsets of I^s .

In this work, we shall define \mathcal{B} as the set of all s -products of intervals

$$\prod_{i=1}^s [u_i, v_i], \quad [u_i, v_i] \pmod{1} \subseteq [0, 1].$$

Often in literature, one considers instead the star-discrepancy $D_N^*(x)$ which is defined by optimizing over the set of all intervals of the form $\prod_{i=1}^s [0, u_i]$.

The definition of discrepancy naturally admits an interpretation in terms of probability: a sequence $x = \{x_n\}_{n \in [N]}$ can be interpreted as a random variable x , that takes the value x_n , as n is the random variable $n \sim U[N]$. Saying that $D_N(x) \leq D_N$ means that for any $B \in \mathcal{B}$ the probability that x_n is contained in B is equal to $\text{vol}(B) + \varepsilon$, where $|\varepsilon| \leq D_N$. In this work, deviating somewhat from standard terminology, we shall often refer to the discrepancy of a distribution x on length N , s -dimensional sequences. In this case $D_N(x)$ will denote

$$D_N(x) = \sup_{B \in \mathcal{B}} \left\{ \left| \mathbb{P}_{x' \sim x, n \sim U[N]}(x'_n \in B) - \text{vol}(B) \right| \right\},$$

⁴applying the union bound over all eigenvalues

Low-discrepancy sequences have much in common with random sampling, or the Monte-Carlo method, in the sense that they visit each cube a number of time that is roughly proportional to its volume, up to a small additive error. Yet, contrary to the Monte-Carlo method, such sequences are *not* random, but only appear to be random in the sense above. Arguably such sequences are most useful in the context of numerical integration: instead of computing an integral of a continuous function f over the s -dimensional cube $[0, 1]^s$ one can replace it with the average value of the function over the points of a low-discrepancy sequence $x = \{x_i\}_{i=1}^N$. The well-known Koksma-Hlawka theorem, then connects the approximation error as a function of the discrepancy $D_N(x)$ as follows:

$$\left| \frac{1}{N} \sum_{i=1}^N f(x_i) - \int_{[0,1]^s} f(z) dz \right| \leq V(f) \cdot D_N^*(x),$$

where $V(f)$ is the bounded-variation of f on $[0, 1]^s$.

There are deterministic s -dimensional sequences $x = \{x_i\}_{i=1}^N$ with discrepancy as low as

$$D_N(x) \leq C \cdot \frac{\log^s N}{N},$$

and matching lower-bounds (up to constant factors) on the smallest possible discrepancy are known for $s = 1$ [Nie92]. Hence, usually one considers low-discrepancy sequences that are very long (N) compared to the dimension (s).

At this point, it may be insightful to consider an example of a low-discrepancy sequence: the well-known van-der Corput sequence [Nie92]: consider the binary expansion of a positive integer

$$\forall n \in [N = 2^b] \quad n = \sum_{i=0}^b \alpha_i 2^i,$$

then the van-der Corput sequence $x = \{x_n\}_{n=1}^N$ is defined as:

$$\forall n \in [N = 2^b] \quad x_n = \sum_{i=0}^b \alpha_i 2^{-i-1}.$$

This sequence has

$$D_N^*(x) \leq C \frac{\log N}{N}.$$

We note that the discrepancy upper-bound decays asymptotically like $O(1/N)$ (assuming small dimension s) whereas the Monte-Carlo method converges more slowly, behaving as $O(1/\sqrt{N})$ - and hence these sequences are often preferred as a method of numerical integration to Monte-Carlo. They are also advantageous compared with purely deterministic methods, like defining a fine-resolution grid, because usually one can increase the length of the quasi-random sequence, and reduce its discrepancy while making use of all previous points of the sequence.

4.2 Some basic facts

We require a Lemma [2.5] due to Niederreiter [Nie92].

Lemma 21. [Nie92] *Small point-wise distance implies similar discrepancy*

Let $x_1, \dots, x_N, y_1, \dots, y_N$ denote two s -dimensional sequences for which $|x_{n,i} - y_{n,i}| \leq \varepsilon$, for all $n \in [N], i \in [s]$. Then the discrepancies of these sequences are related by:

$$|D_N(x_1, \dots, x_N) - D_N(y_1, \dots, y_N)| \leq s \cdot \varepsilon. \quad (1)$$

We prove an additional fact:

Fact 22. Let $x = \{x_n\}_{n \in [N]}$ be a distribution on sequences with discrepancy at most $D_N(x)$, and let $y = \{y_n\}_{n \in [N]}$ denote the following sequence:

$$y_n = x_n + z_n,$$

where $z = \{z_n\}_{n \in [N]}$ is some sequence chosen independently from x . Then

$$D_N(y) = D_N(x).$$

Proof. For each $S \in \mathcal{B}$ we have

$$\mathbb{P}_{y, n \sim U[N]}(y_n \in S) = \mathbb{P}_{x, z, n \sim U[N]}(x_n + z_n \in S) = \int_{[0,1]^s} \mathbb{P}_{x, n \sim U[N]}(x_n \in S - z_n) \cdot \mathbb{P}_{z, n \sim U[N]}(z_n) dz_n$$

By the discrepancy assumption on the sequence x we have that

$$\forall S \in \mathcal{B} \quad |\mathbb{P}_{x, n \sim U[N]}(x_n \in S) - \text{vol}(S)| \leq D_N(x).$$

Therefore by the above

$$\mathbb{P}_{y, n \sim U[N]}(y_n \in S) = \int_{[0,1]^s} (\text{vol}(S) + \varepsilon(z_n)) \cdot \mathbb{P}_{z, n}(z_n) dz_n, \quad |\varepsilon(z_n)| \leq D_N(x).$$

and hence by convexity:

$$\mathbb{P}_{y, n \sim U[N]}(y_n \in S) = \text{vol}(S) + \varepsilon, \quad |\varepsilon| \leq D_N(x),$$

which implies the claim by the definition of discrepancy. □

4.3 The Good Seed Problem

In our context we will be interested in sequences $x = \{x_n\}_{n=1}^N$ of the form

$$x_n = \left\{ \frac{gn}{N} \right\},$$

where $g \in [N]^s$ is some s -dimensional vector, called the *seed* of the sequence. Specifically, the vector g/N will be the vector of eigenvalues of an $n \times n$ Hermitian matrix A whose spectrum $g = \mathcal{L}(A)$ we would like to analyze. Since it is unreasonable to assume that the input matrix has a spectrum that is a good seed, we would like to find a perturbation of the matrix $A' = A + \mathcal{E}$ such that $g' = \mathcal{L}(A')$ has a corresponding sequence, defined as above, with low-discrepancy.

Niederreiter has shown [Nie92] that if g is sampled uniformly on $[N]^s$ then it is a good seed with high probability:

Lemma 23. Let s, N be integers and $g \sim U([N]^s)$. Then

$$\mathbb{P} \left(D_N(x) \leq \frac{\log^s N}{N} \right) \geq 1 - 1/N.$$

For our application we require that $N = \text{poly}(n)$, and $s = 2$, in which case the above discrepancy is sufficiently low for our purposes. Yet, since it requires the normalized seed g/N to be essentially uniform on $[0, 1]^n$, it implies that the corresponding matrix perturbation \mathcal{E} added to A must be very strong - thereby loosing all connection to the input matrix.

4.4 Finding Reasonably-Good Seeds Locally

To bridge the gap between weak-perturbation and low-discrepancy we show a new lemma, which may be of independent interest: it allows to trade-off the extent to which g is random, and the discrepancy of the sequence generated by g . Specifically, we will show that if g/N is uniform on *cubes* of much smaller side-length, i.e. at least $1/\sqrt{N}$, then the resulting sequence has discrepancy $O(\log^s N/\sqrt{N})$. This is the subject of the following lemma:

Lemma 24. *We are given integer N , with prime divisor $M = \Theta(N^a)$ for some constant $a > 0$, and an integer s . Let $g = (g_1, \dots, g_s) \in N^s$, such that each coordinate g_i is independently chosen uniformly on some interval $I_i \subseteq [N]$ of size M . Let $x = x(g) = \{x_n\}_{n=1}^N$ be the following s -dimensional sequence of length N corresponding to residuals of g :*

$$x_n = \left\{ \frac{g \cdot n}{N} \right\}.$$

Then

$$\mathbb{P}_g \left(D_N(x) \leq 2\log^s(M)/\sqrt{M} \right) \geq 1 - 1/\sqrt{M}.$$

Proof. For integers P, s put $C_s^*(P)$ as the set of all vectors in \mathbb{Z}^s with entries in $[-P/2, P/2) \cap \mathbb{Z}$, excluding the all-zero vector. Following Niederreiter [Nie92] we define for each $h \in \mathbb{Z}^s$

$$r(h) = \prod_{i=1}^s r(h_i), \quad (2)$$

where $r(h_i) = \max(1, |h_i|)$. For $g = (g_1, \dots, g_s) \in \mathbb{Z}^s$, we denote:

$$R(g, P) = \sum_{h \cdot g = 0 \pmod{P}, h \in C_s^*(P)} r(h)^{-1}. \quad (3)$$

By Theorem [5.10] of [Nie92] when each g_i is randomly chosen on the entire interval $[P]$, for prime P , then

$$\mathbb{E}_g [R(g, P)] = O \left(\frac{\log^s(P)}{P} \right). \quad (4)$$

Since $R(g, P) \geq 0$ for all vectors g , then

$$\mathbb{P}_g \left(R(g, P) \geq \frac{\log^s(P)}{\sqrt{P}} \right) \leq 1/\sqrt{P}. \quad (5)$$

or

$$\mathbb{P}_g \left(R(g, P) \leq \frac{\log^s(P)}{\sqrt{P}} \right) \geq 1 - 1/\sqrt{P}. \quad (6)$$

Let us use the above equation to upper-bound the discrepancy of $S(g)$. Recall that M is a prime divisor of N , with $M = \Theta(N^a)$. We first observe that:

$$R(g, N) = \sum_{h \cdot g = 0 \pmod{N}, h \in C_s^*(N)} r(h)^{-1} \leq \quad (7)$$

$$\sum_{h \cdot g = 0 \pmod{M}, h \in C_s^*(M)} r(h)^{-1} + \sum_{h \cdot g = 0 \pmod{M}, h \in C_s^*(N), \exists i, \text{ s.t. } |h_i| \geq M} r(h)^{-1} \quad (8)$$

We note that the second term is at most

$$\frac{s}{M} \sum_{h \in \mathbb{Z}^{s-1}} r(h)^{-1} = \frac{s \cdot \log^{s-1}(N)}{M} \quad (9)$$

Regarding the first term, $h \cdot g = 0(\text{mod } M)$ if and only if $h \cdot (g(\text{mod } M)) = 0(\text{mod } M)$, and since each g_i is uniform on some interval of size M , then $g_i(\text{mod } M)$ is uniform on $[M]$. Since M is prime, we can apply equation (6) to the first term with $P = M$. Using this equation and the upper-bound on the second term we have:

$$P_g \left(R(g, N) \leq \frac{2\log^s(M)}{\sqrt{M}} \right) \geq 1 - 1/\sqrt{M}. \quad (10)$$

According to Theorem 5.6 of [Nie92], the discrepancy of the sequence $\{gn/N\}_{n=0}^{N-1}$ is upper-bounded by:

$$D_N(S(g)) \leq \frac{s}{N} + 2^{-s} \cdot R(g, N). \quad (11)$$

Plugging equation (10) into equation (11) implies:

$$P_g \left(D_N(S(g)) \leq \frac{2\log^s(M)}{\sqrt{M}} \right) \geq 1 - 1/\sqrt{M}. \quad (12)$$

□

5 A Filtering Algorithm

In this section we provide the specification of the filtering algorithm, which is the main computational black box of our algorithm. This algorithm accepts an integer m that separates the i -th eigenvalue of a Hermitian matrix A and computes an approximation for the i -th eigenvector, with high probability:

Algorithm 25. Filter(A, m, δ)

1. *Compute parameters:*

$$p = 24n^2 \lceil \ln(1/\delta) \rceil, \zeta = \delta^2 / (2pm).$$

2. **Sample random unit vector:**

Sample a standard complex Gaussian vector v , set $w_0 = v/\|v\|$.

3. **Approximate matrix exponent:**

Compute a ζ Taylor-series approximation of e^{iA} , denoted by \tilde{U} .

4. **Raise to power:**

Compute \tilde{U}^m by repeated squaring.

5. **Generate matrix polynomial:**

Compute $B = \left(\frac{I + \tilde{U}^m}{2} \right)^p$ by repeated squaring.

6. **Filter:**

Compute $w = \frac{B \cdot w_0}{\|B \cdot w_0\|}$.

7. **Decide:**

Set $z = A \cdot w$, $i_0 = \arg \max_{i \in [n]} |w_i|$ and compute $c = z_{i_0}/w_{i_0}$. If

$$\|A \cdot w - c \cdot w\| \leq 3\delta\sqrt{n}$$

return w , and otherwise reject.

We now show that if the algorithm is provided with an integer m that separates the k -th eigenvalue of A in the sense defined in Definition 6, then the output is close to the k -th eigenvector of A .

Theorem 26. Let $\delta \leq n^{-10}$ and $\alpha = \sqrt{\ln(1/\delta)}$. We are given an $n \times n$ Hermitian matrix A with eigenvalues $\{\lambda_i\}_{i \in [n]}$ and a corresponding orthonormal eigenbasis $\{v_i\}_{i \in [n]}$. Additionally, we are provided an integer m that separates k in A , w.r.t. $B_{in}(\alpha), B_{out}$, in the sense of Definition 5. Let $w = \text{Filter}(A, m, \delta)$. Then

$$\mathbb{P}(\|w - v_k\| \leq \delta) \geq 1 - 3n^{-3}.$$

The algorithm has boolean complexity at most $2cn^\omega \cdot \log(2p^2m^2/\delta^2)$, and runs in parallel time $O(\log^2(n))$.

Proof. Let $\{\tau_\ell\}_{\ell \in [n]}$ denote the set of eigenvalues of \tilde{U} . Since \tilde{U} is a polynomial in A (truncated Taylor series) then $\{v_\ell\}_{\ell \in [n]}$ is also an orthonormal basis for \tilde{U} . Since in addition $\|\tilde{U} - e^{iA}\| \leq \zeta$ then

$$\forall \ell \in [n] \quad |\tau_\ell - e^{i\lambda_\ell}| \leq \zeta. \quad (13)$$

Let $w' = B \cdot w_0$ and denote

$$w_0 = \sum_{\ell \in [n]} \beta_\ell v_\ell, \quad \text{and} \quad w' = \sum_{\ell \in [n]} \alpha_\ell v_\ell.$$

Since A, \tilde{U} share the same basis of eigenvectors, then by the definition of the matrix B the coefficients α_ℓ, β_ℓ are related by:

$$|\alpha_\ell|^2 = |\beta_\ell|^2 \cdot \left| \frac{1 + \tau_\ell^m}{2} \right|^{2p}.$$

So by Equation 13

$$\frac{|\alpha_\ell|^2}{|\beta_\ell|^2} \geq \left| \frac{1 + e^{im\lambda_\ell}}{2} \right|^{2p} - 2pm\zeta.$$

Since m separates k then $\{m\lambda_k\} \in \mathcal{B}_{in}$, and for all $\ell \neq k$ we have $\{m\lambda_\ell\} \notin \mathcal{B}_{out}$. Thus, for $\ell = k$:

$$\frac{|\alpha_k|^2}{|\beta_k|^2} \geq \left| \left(1 + \cos(1/2n\sqrt{\ln(1/\delta)}) \right) \right|^p - 2pm\zeta$$

Using Claim 30

$$\geq \left(1 - \frac{1}{4n^2 \ln(1/\delta)} \right)^p - 2pm\zeta \geq \frac{1}{2e^6}. \quad (14)$$

On the other hand, for all $\ell \neq k$ we have:

$$\frac{|\alpha_\ell|^2}{|\beta_\ell|^2} \leq \left| \frac{1 + e^{im\lambda_\ell}}{2} \right|^{2p} + 2pm\zeta.$$

so since m separates k then:

$$\leq |(1 + \cos(1/2n))|^p + 2pm\zeta$$

Using Claim 30

$$\leq (1 - 1/12n^2)^{24n^2 \ln(1/\delta)} + 2pm\zeta \leq e^{-2 \ln(1/\delta)} + 2pm\zeta \leq 2\delta^2. \quad (15)$$

By Fact 29 for any $\varepsilon = 1/\text{poly}(n)$ there exists a constant $c > 0$ such that

$$P(\forall i, j \quad |\beta_j| \leq c|\beta_i| \sqrt{\ln(1/\varepsilon)}/\varepsilon) \geq 1 - 3n\varepsilon.$$

Choose $\varepsilon = n^{-4}$. Then by Equations 14 and 15:

$$P\left(\forall \ell \neq k \quad \frac{|\alpha_\ell|^2}{|\alpha_k|^2} \leq c^2(2\delta^2) \cdot (2e^6) \cdot 4 \ln n \cdot n^8\right) \geq 1 - 3n^{-3}.$$

and so for $\delta \leq n^{-10}$ there exists $\eta \in \mathbb{C}$, $|\eta| = 1$ such that

$$\left\| \frac{w'}{\|w'\|} - \eta \cdot v_k \right\|^2 \leq \frac{1}{|\alpha_k|^2} \sum_{j \neq k} |\alpha_j|^2 \leq 16c^2 n^9 \ln n \delta^2 e^6 < \delta.$$

for sufficiently large n . Using Claim 28 we conclude that w.p. at least $1 - 3n^{-3}$ over choices w_0 , the criterion is met and the algorithm returns a vector $w = w'/\|w'\|$ satisfying the equation above.

Arithmetic run-time: The approximation of e^{iA} by \tilde{U} requires, using Fact 31 a time at most

$$cn^\omega \log(1/\zeta) = cn^\omega \cdot \log(2pm/\delta^2).$$

Next, the repeated powering of \tilde{U} to a power m requires time at most: $cn^\omega \lceil \log(m) \rceil$ and the repeated powering of B to the power p requires time at most: $cn^\omega \lceil \log(p) \rceil$. Hence the total complexity is: $4cn^\omega \cdot \log(pm/\delta^2)$.

Depth complexity: Each matrix product can be carried out in depth $\log(n)$. Each of steps 3 to 6 involves at most $\log(m) + \log(p)$ sequential matrix multiplications. The sorting algorithm can be computed in depth $O(\log(n))$ by Fact 11. Hence the depth complexity of the entire circuit is at most $\log(n) \cdot (\log(m) + \log(p)) + O(\log(n)) = O(\log^2(n))$.

We conclude the proof of the theorem by showing stability:

Claim 27. *Under the assumption of Theorem 26 the algorithm is log-stable.*

Proof: Consider the arithmetic operations involved in computing the filtering algorithm:

1. Generating an approximation \tilde{U} of e^{iAm} as a truncated Taylor series.
2. Raising \tilde{U} to a power $m \in [M]$.
3. Computing $((I + \tilde{U})/2)^p$.
4. Normalizing $Bw_0/\|Bw_0\|$.

Consider an arithmetic circuit C implementing the above, and the circuit $D = D(C, t)$ - the discretization of C to t bits of precision modeled as follows: after each arithmetic step, the result is rounded to the nearest value of 2^{-t} . Consider all steps except division. A is δ -separated so in particular $\|A\| \leq 1$. Thus, whenever we multiply two matrices at any of the steps above both have norm at most 1. Hence, at each rounding step the error is increased by at most $\sqrt{n}2^{-t}$. Finally, considering the final division step, we observe that since m separates k , then by Equation 14 we have $\|Bw_0\| \geq 1/(2e^6)$. This implies that the total error is at most $\sqrt{n}(p + M) \cdot 2^{-t} \cdot 2e^6$. Since M, p are both polynomial in n then for any $\delta = 1/\text{poly}(n)$ the error is at most δ for some $t = O(\log(1/\delta))$. □

5.1 Supporting Claims

Claim 28. Suppose that $\|w - v\| \leq \delta$ for some unit eigenvector v of A , and $\delta \leq 1/4$. Let $z = A \cdot w$, and i_0 denote $i_0 = \arg \max_{i \in [n]} |w_i|$. Let $c = z_{i_0}/w_{i_0}$. Then

$$\|A \cdot w - c \cdot w\| \leq 3\delta\sqrt{n}.$$

Proof. Write $w = v + \mathcal{E}$, $\|\mathcal{E}\| \leq \delta$. Since $\|A\| \leq 1$ then $z = Aw = \lambda_v \cdot v + \mathcal{E}'$, where $\|\mathcal{E}'\| \leq \delta$. Therefore

$$z_{i_0} = \lambda_v \cdot v_{i_0} + \mathcal{E}'_{i_0} = \lambda_v \cdot (w_{i_0} - \mathcal{E}_{i_0}) + \mathcal{E}'_{i_0},$$

So

$$z_{i_0} = \lambda_v w_{i_0} + \mathcal{E}'', |\mathcal{E}''| \leq 2\delta.$$

Since $|w_{i_0}| \geq 1/\sqrt{n}$ then:

$$c = \frac{z_{i_0}}{w_{i_0}} = \lambda_v + \zeta, |\zeta| \leq 2\delta\sqrt{n}.$$

Hence

$$\|A \cdot w - c \cdot w\| \leq \|\lambda_v \cdot v + \mathcal{E}' - (\lambda_v + \zeta) \cdot (v + \mathcal{E})\| = \|\mathcal{E}' - (\lambda_v + \zeta)\mathcal{E} - \zeta v\|$$

which by the triangle inequality is at most

$$\delta + (1 + 2\delta\sqrt{n})\delta + 2\delta\sqrt{n} \leq 3\delta\sqrt{n},$$

for sufficiently large n . □

Fact 29. Random unit vectors have well-balanced entries

Let $\{v_i\}_{i \in [n]}$ be some orthonormal basis of \mathbb{C}^n , $0 < \varepsilon = 1/\text{poly}(n)$, and $v \in \mathbb{C}^n$ a uniformly random complex unit vector. For any $i \in [n]$ let $\alpha_i = |\langle v, v_i \rangle|$. For any $\varepsilon = 1/\text{poly}(n)$ there exists a number $c_1 > 0$ independent of n , such that

$$\mathbb{P}\left(\forall i, j \quad |\alpha_i|/|\alpha_j| \leq c_1 \sqrt{\ln(1/\varepsilon)/\varepsilon}\right) \geq 1 - 3n\varepsilon.$$

Proof. Sample a random unit vector by sampling a standard Gaussian vector and normalizing to unity. Let $x = (x_1, \dots, x_n)$ where $x_i \sim \mu(0, 1)$, and all are i.i.d. Since the Gaussian measure is invariant under conjugation by a unitary matrix we can assume w.l.o.g. that $v_i = e_i$ for each i , in which case the α_i 's are i.i.d. $\alpha_i \sim \mu(0, 1)$. By the Gaussian measure for every $c_2 > 0$ there exists $c_1 > 0$ such that

$$\mathbb{P}(|\alpha_i| \geq c_1 \sqrt{\ln n}) \leq n^{-c_2}.$$

Therefore, by the union bound

$$\mathbb{P}(\forall i \quad |\alpha_i| \geq c_1 \sqrt{\ln n}) \leq n^{-c_2+1}.$$

On the other hand, for any $\alpha_i, 0 < \varepsilon < 1/n$, we have

$$\mathbb{P}(|\alpha_i| \leq \varepsilon) \leq 2\varepsilon.$$

So by the union bound

$$\mathbb{P}(\forall i \in [n] \quad |\alpha_i| \leq \varepsilon) \leq 2n\varepsilon.$$

Set $c_2 = \ln_n(1/\varepsilon)$. Since $\varepsilon = n^{-k}$ for some constant k , then $c_2 = k$ is also a constant. Then there exists a constant c_1 such that:

$$\mathbb{P}\left(\forall i, j \quad |\alpha_i|/|\alpha_j| \geq c_1 \sqrt{\ln n/\varepsilon}\right) \leq 2n\varepsilon + n^{-c_2+1} = 3n\varepsilon,$$

Since normalization does not change these ratios, then the maximal ratio between the components of x , is the same as the maximal ratio between the components of $x/\|x\|$. This implies the proof. □

Claim 30. $\forall \theta \in [-0.01, 0.01] \quad 1 - \frac{\theta^2}{2} \leq \frac{1 + \cos(\theta)}{2} \leq 1 - \frac{\theta^2}{3}.$

Proof. Follows by truncating the Taylor series of $\cos(x)$ to second order. \square

Fact 31. Efficient approximation of exponentiated matrix

Given a Hermitian $n \times n$ matrix A , $\|A\| \leq 1$, and error parameter $\varepsilon > 0$, a Taylor approximation of e^{iA} , denoted by \tilde{U}_A can be computed in time $cn^\omega \log(1/\varepsilon)$ and satisfies $\|e^{iA} - \tilde{U}_A\| \leq \varepsilon$.

Proof. Put $s = \log(1/\varepsilon)$ and consider the Taylor approximation of e^{iA} up to the first s terms:

$$\tilde{U}_A := \sum_{m=0}^{s-1} \frac{(iA)^m}{m!}$$

Then the approximation error can be bounded as:

$$\|e^{iA} - \tilde{U}_A\| \leq \sum_{m=s}^{\infty} \frac{\|A^m\|}{m!} \leq \sum_{m=s}^{\infty} \frac{\|A\|^m}{m!} \leq \sum_{m=s}^{\infty} \frac{1}{m!} \leq 2^{-s} \leq \varepsilon, \quad (16)$$

where we have used the fact that $\|A\| \leq 1$. The complexity of the approximation is comprised of $\log(1/\varepsilon)$ matrix products for a total of $cn^\omega \log(1/\varepsilon)$. \square

6 Sampling Separating Integers

In this section we show our main technical tool: which is that perturbing a δ -separated Hermitian matrix A by a Gaussian matrix of a carefully calibrated variance, results in a corresponding sequence of residuals $S(A)$ having low-discrepancy, at least for pair-wise variables, which in turn implies that we can separate each eigenvalue of A almost uniformly:

Theorem 32. Let A be a δ -separated $n \times n$ PSD matrix, \mathcal{E} GUE, $\zeta \leq \min\{\delta^{13}, n^{-50}\}$, and $4 < \alpha \leq n$. For any $M \geq \zeta^{-1.6}$ we have:

$$\forall k \in [n] \quad \mathbb{P}_{\mathcal{E}, m \sim U[M]} (m \text{ separates } k \text{ in } A + \zeta \cdot \mathcal{E} \text{ w.r.t. } B_{in}(\alpha), B_{out}) \geq 1/(5\alpha n)$$

6.1 Additive Perturbation

Definition 33. (σ, ε) -normal vector

Let $v = (v_1, \dots, v_n)$ be a vector of n random variables. v is said to be (σ, ε) -normal, if each $v_i \sim x_i + y_i$ with $x_i \sim \mu(\lambda_i, \sigma^2)$ for some $\lambda_i \in \mathbb{R}$, x_1, \dots, x_n are independent, and $|y_i| \leq \sigma\varepsilon$, for all i .

By our definitions above, Gaussian perturbation of a matrix with well-separated eigenvalues results in a (σ, ε) -normal vector as follows:

Fact 34. Perturbation of well-separated matrices

Let A be an $n \times n$ α_{min} -separated Hermitian matrix with eigenvalues $\lambda_1 \geq \lambda_2 \dots \geq \lambda_n$, where $\alpha_{min} \geq \varepsilon$, and $\varepsilon > 0$ is some constant. Let \mathcal{E} be GUE, and $A' = A + \varepsilon^L \cdot \mathcal{E}$, where $L \geq 2$. Then w.p. at least $1 - n \cdot 2^{-n}$ the vector of eigenvalues of A' ($\lambda'_1, \dots, \lambda'_n$) is $(\varepsilon^L, cn\varepsilon^{L-1})$ -normal, for some constant $c > 0$.

Proof. We invoke Corollary 16 choosing ε as ε^L and δ as ε . We get:

$$\forall i \in [n] \quad \lambda'_i = (1 - \alpha) \cdot \mu(\lambda_i, \varepsilon^{2L}) + \alpha \cdot \mathcal{D} + \zeta_i, \quad |\zeta_i| \leq 16cn \cdot \varepsilon^{2L-1}, 0 \leq \alpha \leq 2^{-n}.$$

Choosing $\sigma = \varepsilon^L$ implies the each λ'_i w.p. at least $1 - 2^{-n}$ is sampled according to:

$$\lambda'_i = \mu(\lambda_i, \varepsilon^{2L}) + \zeta_i, \quad |\zeta_i| \leq \sigma \cdot 16cn \varepsilon^{L-1}.$$

Taking the union bound over all $i \in [n]$ then implies the proof. \square

6.2 Moderately Low-Discrepancy

Lemma 35. Low-discrepancy sequence from almost normal vectors

Let $B > 0$, and $v = (v_1, \dots, v_n)$ be some (σ, ε) -normal vector, for $\sigma = n^{-B}, \varepsilon \leq n^{-0.9B}$. There exists $M \leq n^{1.6B}$ such that for any $S = \{i_1, \dots, i_s\} \subseteq [n]$, $|S| = s$ the distribution on s -dimensional sequence of length M :

$$V_s \equiv \{(\{m \cdot v_{i_1}\}, \dots, \{m \cdot v_{i_s}\})\}_{m \in [M]}$$

satisfies

$$D_M(V_s) \leq 4\log^s(n) \cdot n^{-0.1B}.$$

Proof. Let P be the minimal prime at least $n^{0.3B}$, and put $M = P^5$. By Bertrand's postulate, for sufficiently large n we have that $M = P^5 \leq n^{1.51B} \leq n^{1.6B}$. For any $z \in [0, 1)$ let z^M be the number closest to z in the grid m/M , $m \in [M]$.

Removal of non-independent component. Since v is (σ, ε) -normal then $v_i = X_i + Y_i$, where $X_i \sim (\eta_i, \sigma^2)$, $|Y_i| \leq \varepsilon\sigma$, and the X_i 's are independent. Let V_S^X denote the sequence generated by taking only the X component of the seed vector v , i.e.:

$$V_S^X \equiv \{(\{m \cdot X_{i_1}\}, \dots, \{m \cdot X_{i_s}\})\}_{m \in [M]} \quad (17)$$

Fact 36.

$$D_M(V_S) \leq D_M(V_S^X) + s \cdot n^{-0.2B}$$

Proof. Consider the r.v.'s X_i, Y_i . By our assumption

$$\forall i \in [n] \quad |Y_i| \leq \sigma\varepsilon = n^{-1.9B}. \quad (18)$$

Thus:

$$\forall m \in [M], i \in [n] \quad |\{mv_i\} - \{mX_i\}| \leq m \cdot n^{-1.9B} \leq Mn^{-1.9B} \leq n^{-0.3B} \quad (19)$$

By Lemma 21, we can conclude that the discrepancy of our target sequence V_S follows tightly the discrepancy of V_S^X :

$$D_M(V_S) \leq D_M(V_S^X) + s \cdot n^{-0.3B} \quad (20)$$

\square

Reducing Gaussian measure to uniform measure Consider the vector derived by truncating each coordinate of the vector $(X_{i_1}, \dots, X_{i_s})$ to the nearest point on the M -grid:

$$X^M = (X_{i_1}^M, \dots, X_{i_s}^M) = (\lfloor MX_{i_1} \rfloor / M, \dots, \lfloor MX_{i_s} \rfloor / M).$$

Consider the discrepancy of the distribution on s -dimensional sequences formed by taking integer multiples of X^M . We claim:

Fact 37.

$$P_v \left(D_M(V_S^{X,M}) \leq \log^s(n) \cdot n^{-0.1B} \right) \geq 1 - 3n^{-0.1B},$$

Proof. In Fact 40 choose as parameter $m = n^{0.2B+2}$. We get that w.p. at least $1 - 2n^2/m = 1 - 2n^{-0.2B}$ each X_i samples a convex mixture of variables $\{w_j\}_{j \in [m]}$ where

$$w_j \sim U(I_j), |I_j| = \sigma/m = n^{-1.2B-2} \quad (21)$$

Hence, w.p. at least $1 - 2n^{-0.2B}$ for all $i \in [n]$, the variable $M \cdot \{X_i^M\}$ is a convex mixture of uniform random variables on intervals $I_j \subseteq [M]$, where

$$|I_j| \geq \frac{\sigma M}{m} \geq n^{1.5B} \cdot n^{-1.2B-2} \geq M^{0.2}. \quad (22)$$

We apply Lemma 24 to the sequence of residuals of integer multiples, with the seed X^M :

$$V_S^{X,M} \equiv (\{mX_1^M\}, \dots, \{mX_s^M\})_{m \in [M]}. \quad (23)$$

The lemma requires that each variable be distributed as:

$$MX_i^M \sim U[\mathcal{I}],$$

where \mathcal{I} is some interval of $[M]$, for integer $M > 1$ satisfying:

$$|\mathcal{I}| \geq P, \quad P \text{ prime}, \quad P \geq N^a, a > 0.$$

By our choice of parameters M has a prime divisor P equal to $M^{0.2} = P$. Hence, by Equation 22 we can satisfy the assumption of the lemma by choosing the parameters N, M, a for Lemma 24 as follows:

$$N = M, M = P, a = 0.2. \quad (24)$$

Hence, by Lemma 24, and accounting for the Gaussian-to-uniform approximation error we get:

$$P_v \left(D_M(V_S^{X,M}) \leq 2\log^s(n) \cdot n^{-0.1B} \right) \geq 1 - n^{-0.1B} - 2n^{-0.2B} \geq 1 - 3n^{-0.1B}. \quad (25)$$

□

Treating the residual w.r.t. the M -grid Define: the truncation error

$$\forall i \in [s] \quad r_i := X_i - X_i^M.$$

In Fact 36 we analyzed the error Y_i whose magnitude is negligible even w.r.t. $1/M$, and can thus be disregarded for any element of the sequence V_S . Unlike this, the residual error r_i cannot be disregarded because when multiplied by integers uniformly in $[M]$ it assumes magnitude $\Omega(1)$. Thus, it requires a different treatment.

Corollary 38.

$$P_v \left(D_M(V_S^X) \leq 2\log^s(n) \cdot n^{-0.1B} \right) \geq 1 - 4n^{-0.1B}$$

Proof. Express the i -th element of the sequence using r_i :

$$\forall i \in [s] \quad \{X_i \cdot m\} = \{(X_i^M + r_i) \cdot m\} = \{\{mX_i^M\} + \{mr_i\}\} \quad (26)$$

Let E denote the event in which X_i is sampled according to $w_j \sim U[\mathcal{I}_j]$ where w_j is at distance at least $1/M$ from either one of the edges of \mathcal{I}_j . Conditioned on E , the random variables r_i and X_i^M are independent for all $i \in [s]$ so conditioned on E we also have

$$\forall i \in [s] \quad \{mX_i^M\}, \{mr_i\} \text{ are independent.}$$

$V_S^{X,M}$ is a distribution on sequences formed by sampling the initial seed $\{X_i^M\}$, and V_S^X is a distribution on sequences formed by adding to $V_S^{X,M}$ a random sequence formed by sampling the independently seed $\{r_i\}_{i \in [s]}$, and then generating the length- M sequence

$$\{(mr_1, \dots, mr_s)\}_{m \in [M]}.$$

Then since conditioned on E $\{mX_i^M\}, \{mr_i\}$ are independent for all i , then by Fact 22 we have:

$$D_M(V_S^X | E) = D_M(V_S^{X,M})$$

and so by Fact 37

$$\mathbb{P}_v(D_M(V_S^X | E) \leq \log^s(n) \cdot n^{-0.1B}) \geq 1 - 3n^{-0.1B}, \quad (27)$$

By Equation 22 the probability of E is at least:

$$\mathbb{P}_v(E) \geq 1 - |\mathcal{I}_j|/(2M) \geq 1 - M^{0.2}/(2M) \geq 1 - n^{-B}.$$

Thus:

$$\mathbb{P}_v(D_M(V_S^X) \leq \log^s(n) \cdot n^{-0.1B}) \geq 1 - 3n^{-0.1B} - \mathbb{P}(E) \geq 1 - 4n^{-0.1B}. \quad (28)$$

□

Conclusion of proof:

By Corollary 38 we have

$$\mathbb{P}_v(D_M(V_S^X) \leq 2\log^s(n) \cdot n^{-0.1B}) \geq 1 - 4n^{-0.1B}$$

and by Fact 36 we have

$$D_M(V_S) \leq D_M(V_S^X) + s \cdot n^{-0.2B}$$

Thus by the union bound:

$$\mathbb{P}_v(D_M(V_S) \leq 2\log^s(n) \cdot n^{-0.1B} + s \cdot n^{-0.2B}) \geq 1 - 4n^{-0.1B}$$

thus:

$$\mathbb{P}_v(D_M(V_S) \leq 3\log^s(n) \cdot n^{-0.1B}) \geq 1 - 4n^{-0.1B}$$

Hence for all but a measure $4n^{-0.1B}$ of sampled vectors v , the resulting sequence has discrepancy at most $3\log^s(n)n^{-0.1B}$. Since the discrepancy measures the worst-case additive error for any set this implies that:

$$D_M(V_S) \leq 3\log^s(n)n^{-0.1B} + 4n^{-0.1B} \leq 4\log^s(n)n^{-0.1B}$$

□

6.3 Approximate Pairwise Independence

Lemma 39. Let $\bar{\lambda} = (\lambda_1, \dots, \lambda_n) \in [0, 1]^n$ and M a positive integer that satisfy:

$$\forall i \neq j \quad D_M(\{(m\lambda_i, m\lambda_j)\}_{m \in [M]}) \leq \zeta, \quad \zeta \leq n^{-4}$$

Let $4 < \alpha \leq n$. For each $k \in [n]$ w.p. at least $1/(5\alpha n)$ over choices of $m \sim U[M]$ the sampled sequence m separates k w.r.t. $B_{in}(\alpha), B_{out}$.

Proof. Let E_i denote the following event:

$$E_i := x_i \in B_{in} \wedge \forall j \neq i \quad x_j \notin B_{out}$$

We want to show that

$$\forall i \in [n] \quad \mathbb{P}(E_i) \geq \frac{1}{5\alpha n}.$$

Let s denote the number of x_j 's not in B_{out} :

$$s = |\{j \mid j \neq i \quad x_j \notin B_{out}\}|$$

Then under this notation we have:

$$\mathbb{P}(E_i) = \mathbb{P}(s = n - 1 \mid x_i \in B_{in}). \quad (29)$$

Consider the conditional expectation:

$$\mathbf{E}[s \mid x_i \in B_{in}]$$

By linearity of expectation:

$$\mathbf{E}[s \mid x_i \in B_{in}] = \sum_{j \neq i} \mathbb{P}[x_j \notin B_{out} \mid x_i \in B_{in}]. \quad (30)$$

Considering each summand separately:

$$\mathbb{P}(x_j \notin B_{out} \mid x_i \in B_{in}) = \frac{\mathbb{P}(x_j \notin B_{out} \wedge x_i \in B_{in})}{\mathbb{P}(x_i \in B_{in})}$$

Using the pairwise discrepancy assumption, the above is at most:

$$\frac{(1 - |B_{out}|) \cdot |B_{in}| + \zeta}{|B_{in}| - \zeta} \leq 1 - |B_{out}| + 2\zeta\alpha n = \frac{1}{2n} + 2\alpha\zeta n \leq \frac{0.51}{n}$$

and so by Equation 30

$$\mathbf{E}[s \mid x_i \in B_{in}] = (n - 1) \cdot \mathbb{P}[x_j \notin B_{out} \mid x_i \in B_{in}] \leq 0.51.$$

The variable $s \mid x_i \in B_{in}$ accepts only integral values, and by Markov's inequality:

$$\mathbb{P}(s \geq 1 \mid x_i \in B_{in}) \leq 0.51$$

Therefore

$$\mathbb{P}(s = 0 \mid x_i \in B_{in}) \geq 0.49.$$

Using again the 1-dimensional discrepancy we have

$$\mathbb{P}(x_i \in B_{in}) \geq \frac{1}{\alpha n} - \zeta \geq \frac{1}{2\alpha n}.$$

Substituting the last two inequalities into Equation 29 yields:

$$\mathbb{P}(E_i) \geq 0.49 \cdot \frac{1}{2\alpha n} \geq \frac{1}{5\alpha n}$$

□

6.4 Proof of Theorem 32

Proof. By assumption A is δ -separated and $\zeta \leq \min\{n^{-50}, \delta^{13}\}$. Consider the perturbed matrix

$$A' = A + \zeta \mathcal{E}.$$

Choose $L = 13$ and $\varepsilon = \zeta^{1/13}$. Then by Fact 34 w.p. at least $1 - n2^{-n}$ all eigenvalues of A' are (σ, ε) -normal with parameters

$$\sigma = \zeta \leq n^{-50}, \varepsilon \leq 16cn\zeta^{12/13} \leq \zeta^{0.9} = \sigma^{0.9}$$

where the last inequality follows because $\zeta \leq n^{-50}$. We assume that this is the case and account for the negligible error at the end. Set now $B = \log_n(1/\sigma)$. Then the eigenvalues of A' are (σ, ε) -normal for $\sigma = n^{-B}$ and $\varepsilon \leq n^{-0.9B}$. Since in addition $4 < \alpha \leq n$ then by Lemma 35 there exists an integer $M \leq n^{1.6B}$ satisfying:

$$\forall S \subseteq [n] \quad D_M(\{m\lambda_S\}) \leq 4\log^s(n)n^{-5} \leq n^{-4}, \quad (31)$$

for sufficiently large n . Hence, by Lemma 39 a random $m \sim U[M]$ separates the k -th eigenvalue of $A + \mathcal{E}$ w.r.t. $B_{in}(\alpha), B_{out}$ w.p at least $1/(5\alpha n)$. \square

6.5 Technical approximations

Fact 40. Approximating a Gaussian by a convex sum of uniform distributions

Let $g = (g_1, \dots, g_n)$ be standard Gaussian vector $g \in \mathbb{R}^n$. Then g is equal to a convex combination of two distributions $\mathcal{D}_u, \mathcal{D}_v$ as follows: $(1-p)\mathcal{D}_U + p \cdot \mathcal{D}_V$, where \mathcal{D}_U is the n -fold distribution of independent variables z_1, \dots, z_n , where each $z_i, |z_i| \leq \sqrt{n}$, and z_i is the convex sum $z_i = \sum_{j=1}^m t_j w_j$ of $m \geq 2n^2$ i.i.d. uniformly distributed variables, with $w_j \sim U[I_j]$, with $|I_j| = 1/m$, and $p \leq 2n^2/m$.

Proof. Partition the interval $[-\sqrt{n}/2, \sqrt{n}/2]$ into $m \cdot \sqrt{n}$ equal intervals I_j , each of size $1/m$, and let p_j denote the point of minimal absolute value in the j -th interval. Set $t_j = \Phi(p_j)$. Consider the real Gaussian PDF $\Phi(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}$. For any pair of neighboring intervals $p_k, p_{k+1}, |p_{k+1}| > |p_k|$, we have

$$|\Phi(p_j) - \Phi(p_{j+1})| \leq \Phi(p_j) \cdot \left(1 - \frac{e^{-(p_{j+1}/m)^2/2}}{e^{-(p_j)^2/2}}\right) \leq 1 - e^{-p_j/m} \leq 1 - e^{-\sqrt{n}/m} \leq \sqrt{n}/m. \quad (32)$$

Then by the above:

$$\forall j \in [m] \quad \max\{|t_j - t_{j-1}|, |t_j - t_{j+1}|\} \leq \sqrt{n}/m,$$

Let $z_i : \mathbb{R} \mapsto [0, 1]$ denote the following function, which is a sum of uniform distributions on the intervals I_j :

$$z_i = \sum_{j=1}^{m\sqrt{n}} t_j U[I_j].$$

Consider the l_1 -distance between g_i and z_i :

$$\int_{x \in \mathbb{R}} |g_i(x) - z_i(x)| dx = \int_{x \in \mathbb{R}} \left| g_i(x) - \sum_{j=1}^{m\sqrt{n}} \Phi(p_j) U[I_j](x) \right| dx$$

By Equation 32, and by bounding the Gaussian tail at \sqrt{n} we establish the upper bound:

$$\leq \frac{1}{m} \sum_{j=1}^{m\sqrt{n}} \max \{|t_j - t_{j+1}|, |t_j - t_{j-1}|\} + 2 \cdot e^{-n} \leq 2n/m.$$

On the other hand, by definition of the points t_j , we have

$$g_i(x) - z_i(x) > 0, \quad \forall x \in \mathbb{R}. \quad (33)$$

Hence each g_i may be written as a convex combination

$$g_i = (1-p) \cdot \hat{z}_i + p \cdot y_i, \quad \hat{z}_i(x) = \frac{z_i(x)}{\int_{\mathbb{R}} z_i(x) dx} \quad p \leq 2n/m.$$

Since the variables g_i are independent then the n -fold distribution of g_1, \dots, g_n , can be written as a convex combination of the of the n -th fold distribution of such i.i.d. variables $\hat{z}_1, \dots, \hat{z}_n$, and another distribution, \mathcal{D}_v occurring, by the union bound, w.p. at most $n \cdot p \leq 2n^2/m$. \square

7 Parallel Algorithm for ASD

The algorithm $\text{Filter}(A, m, \delta)$ described in Section 5 is given an integer m that separates the i -th eigenvalue, and returns an approximation for the i -th eigenvector. In this section, we use this algorithm in a black-box fashion and design a Las-Vegas algorithm for computing the full ASD of a matrix. Essentially, it amounts to running sufficiently many copies of the filtering algorithm in parallel so that all eigenvectors are (coupon) collected with high probability.

Algorithm 41. *Input: $n \times n$ Hermitian matrix A , parameter $\delta \leq 1/n$.*

1. *Compute parameters:*

$$B = \min\{\delta, B^*(\delta/(3\sqrt{n}))\}, \delta' = (\min\{\delta, B\})^{13}/4$$

$$\alpha = \sqrt{\ln(1/\delta')}, M = (\max\{B^{-12}, n^{-50}\})^{1.6}, \mathcal{T} = 60n\alpha \log(n)$$

$$B_{out} = [1 - 1/(2n), -1 + 1/(2n)] \quad \text{and} \quad B_{in} = [-1/(n\alpha), 1/(n\alpha)],$$

2. *Perturb: $A_1 = A + \sqrt{(\delta')^2 + \delta^2/(9n)} \cdot \mathcal{E}$, where \mathcal{E} is GUE.*

3. *Run \mathcal{T} parallel processes of the following procedure*

(a) *Sample $m \sim U[M]$*

(b) *Run Filter (A_1, m, δ') .*

4. *For vector $w = w_k$ sampled at process $i \in [\mathcal{T}]$, compute $z = A \cdot w$, $i_0 = \arg \max_{i \in [n]} |w_i|$ and $\tilde{\lambda}_k = z_{i_0}/w_{i_0}$.*

5. *Sort the values $\tilde{\lambda}_i$: assume $\tilde{\lambda}_1 \leq \dots \leq \tilde{\lambda}_{\mathcal{T}}$. Initialize: $\gamma = \tilde{\lambda}_1$, $\mathcal{D} = \Phi$. Iterate over all $i = 1, \dots, \mathcal{T}$. At each step i : if $|\gamma - \tilde{\lambda}_i| \geq B/4$ then add $\mathcal{D} \rightarrow \mathcal{D} \cup \{w_i\}$, and set $\gamma = \tilde{\lambda}_i$.*

We now state our main theorem:

Theorem 42. For any Hermitian matrix $0 \preceq A \preceq 0.9I$, and $\delta \leq n^{-10}$ there exists an $\text{RNC}^{(2)}$ algorithm computing $\text{ASD}(A, \delta)$, in boolean complexity $\tilde{O}(n^{\omega+1})$. The algorithm is log-stable.

Proof.

Correctness:

Let $\mathcal{E}_1, \mathcal{E}_2$ be independent GUE matrices, and set

$$A_0 = A + (\delta/(3\sqrt{n})) \cdot \mathcal{E}_1.$$

By Lemma 18 and definition 19 the parameter $B \leq B^*$ above satisfies:

$$\mathbb{P}(A_0 \text{ is } B \text{ separated}) \geq \mathbb{P}(A_0 \text{ is } B^* \text{ separated}) \geq 0.99.$$

Assume that this is the case, and account for the error at the end. Consider now a small-scale perturbation of A_0 :

$$A_1 = A_0 + \delta' \cdot \mathcal{E}_2. \quad (34)$$

Since $\mathcal{E}_1, \mathcal{E}_2$ are independent the matrix A_1 is a perturbation of A as follows:

$$A_1 = A + \sqrt{(\delta')^2 + (\delta/(3\sqrt{n}))^2} \cdot \mathcal{E}, \quad (35)$$

for GUE matrix \mathcal{E} . Hence the matrix A_1 used by the algorithm starting at step 3 is a δ' -perturbation of a B -separated matrix A_0 .

We now invoke Theorem 32 w.r.t. A_0 . A_0 is separated with parameter B , and perturbed by GUE with standard deviation $\zeta = \delta' \leq B^{13}$. We also have $\delta \leq n^{-10}$ which means in particular that the parameter ζ used in the statement of Theorem 32 satisfies $\zeta \leq n^{-50}$. Finally we have $2 < \alpha = O(\sqrt{\ln(n)}) = o(n)$. Therefore:

$$\forall k \in [n] \quad \mathbb{P}_{\mathcal{E}_2, m \sim U[M]}(m \text{ separates } k \text{ in } A_1 = A_0 + \delta' \mathcal{E}_2 \text{ w.r.t. } B_{in}(\alpha), B_{out}) \geq 1/(5\alpha n)$$

Let $\{v_i\}_{i=1}^n$ denote an orthonormal basis for A_1 , and let λ_i denote the eigenvalues of A_1 .

Conditioned on sampling A_1, m such that m separates k in A_1 , we invoke Theorem 26 for $w = \text{Filter}(A, m, \delta')$. By our choice of parameters we have that $\delta' \leq n^{-10}$, and $\alpha = \sqrt{\ln(1/\delta')}$. Hence, the conditions of the theorem are met and so the output vector w satisfies:

$$\mathbb{P}_{w_0}(\|w - v_k\| \leq \delta') \geq 1 - 3n^{-3}.$$

Thus, by the union bound we have

$$\forall k \in [n] \quad \mathbb{P}_{\mathcal{E}_2, m, w_0}(\|w - v_k\| \leq \delta') \geq \frac{1}{5n\alpha} - 3n^{-3} \geq \frac{1}{6n\alpha}$$

Therefore, by the coupon collector's bound the probability that $\mathcal{T} = 60n\alpha \log(n)$ parallel copies sample an approximation for each $k \in [n]$ is at least

$$1 - 6/60 - 0.01 = 0.89$$

Assume that this is the case. Let $\{w_i\}_{i=1}^{\mathcal{T}}$ denote the output set of vectors from all \mathcal{T} parallel copies. We now want to show that each output vector is a valid approximate eigen-vector. Each output vector satisfies by the stopping criterion:

$$\exists c \quad \left\| \tilde{\lambda}_i \cdot w_i - A_1 \cdot w_i \right\| \leq 3\delta' \sqrt{n} \leq B/200,$$

where the last inequality is for sufficiently large n , and $\tilde{\lambda}_i$ is the algorithm's approximation of the i -th eigenvalue. By Fact 15 since A_0 is B -separated then

$$P(A_1 \text{ is } B/2 \text{ separated}) \geq 1 - 2^{-n}. \quad (36)$$

Assuming this is the case, $\delta' \sqrt{n}$ is negligibly smaller than the separation of A_1 - so by Fact 44 the value $\tilde{\lambda}_i$ above must satisfy:

$$\exists k \in [n] \quad |\tilde{\lambda}_i - \lambda_k| \leq B/10.$$

and there exists some eigenvector v_k of λ_k such that

$$\|w_i - v_k\| \leq B/30.$$

This implies by the triangle inequality that for any $j \neq i, l \neq k$ for which $|\tilde{\lambda}_j - \lambda_l| \leq B/10$ we have

$$|\tilde{\lambda}_i - \tilde{\lambda}_j| > B/2 - 2B/10 > B/4.$$

hence $\tilde{\lambda}_i$ and $\tilde{\lambda}_j$ are classified to different eigenvalue bins at step 5. We conclude that the set of values $\{\tilde{\lambda}_i\}_{i=1}^T$ satisfies that

- For every $k \in [n]$ there exists $j \in \mathcal{T}$ such that: $|\lambda_k - \tilde{\lambda}_j| \leq B/10$ and there exists a sampled vector w_j , and a unit eigenvector v_k of λ_k such that $\|w_j - v_k\| \leq B/30$.
- For every $j \in \mathcal{T}$ there exists a unique $k \in [n]$ such that $|\lambda_k - \tilde{\lambda}_j| \leq B/10$

Therefore, at the end of step 5 we have $|\mathcal{D}| = n$ and

$$\forall k \in [n] \quad \exists w \in \mathcal{D} \quad \|w - v_k\| \leq B/10.$$

Hence, with probability at least $0.89 - 2^{-n} \geq 0.85$ the algorithm returns a database $\mathcal{D} = \{w_1, \dots, w_n\}$, such that there exists an orthonormal basis $\{v_1, \dots, v_n\}$ of A_1 for which

$$\forall k \in [n] \quad \|v_k - w_k\| \leq B/10 \leq \delta/10.$$

In that case \mathcal{D} is an $\text{ASD}(A_1, \delta/10)$. On the other and, by Equation 35 we have that w.p. $1 - 2^{-n}$ A_1 satisfies:

$$\|A - A_1\|^2 \leq (\delta/3)^2.$$

Therefore w.p. at least 0.84 the database \mathcal{D} is also $\text{ASD}(A, \sqrt{(\delta/10)^2 + (\delta/3)^2})$ which is in particular an $\text{ASD}(A, \delta)$.

Run time:

By Theorem 26 the parallel time of each copy of $\text{Filter}(A, m, \delta')$ is at most $O(\log^2(n))$, and arithmetic complexity $\tilde{O}(n^\omega)$. Hence, this is the depth complexity of all \mathcal{T} parallel copies of the circuit for $\text{Filter}(A, m)$. The sorting step 5 can be implemented in depth $O(\log(n))$ by Fact 11, using $O(n)$ processors. Hence the complete algorithm runs in parallel time $O(\log^2(n))$ in arithmetic complexity $\tilde{O}(n^{\omega+1})$.

Finally, we show that the entire algorithm can be implemented in $\log(n)$ -precision:

Proposition 43. *The algorithm 41 is log-stable. In particular it runs in boolean complexity $\tilde{O}(n^{\omega+1})$.*

Proof. Each parallel process is log-stable by Claim 27. Hence, to show that the entire algorithm is log-stable, it is sufficient to show that for a Gaussian perturbation truncated to $t = O(\log(n))$ bits of precision, the statement of Theorem 32 still applies - namely a uniformly

random $m \sim U[M]$ separates each eigenvalue, with high probability, for the parameter M set in the algorithm. To do that, write

$$\mathcal{E}_t = \mathcal{E} + D,$$

where \mathcal{E} is the non-truncated GUE matrix, and D is some matrix sampled from a distribution such that $|D_{i,j}| \leq 2^{-t}$ for all i, j . Following Equation 34 define

$$A_1^{(t)} \equiv A_0 + \mathcal{E} + D = A_1 + D,$$

which is then used by the parallel process of $\text{Filter}(A_1^{(t)}, m, \delta')$. By Equation 36 w.p. at least $1 - 2^{-n}$ the matrix A_1 is separated with parameter at least $B/2$. Hence, for sufficiently large $t = O(\log(n))$, such that $2^t > B$ and using again Fact 15 we have for each $i \in [n]$:

$$\left| \lambda_i(A_1) - \lambda_i(A_1^{(t)}) \right| \leq 2\sqrt{n} \cdot 2^{-t}.$$

Hence by Lemma 21 if

$$(\{m\lambda_{i_1}(A_1)\}, \dots, \{m\lambda_{i_2}(A_1)\})_{m \in [M]},$$

is a 2-dimensional sequence with discrepancy D_M , then the discrepancy of the noisy sequence: $(\{m\lambda_{i_1}(A_1^{(t)})\}, \dots, \{m\lambda_{i_2}(A_1^{(t)})\})_{m \in [M]}$ is at most $D_M + 2 \cdot M \cdot \sqrt{n} \cdot 2^{-t}$. This implies that the pairwise discrepancy error increases by an additive error at most $2M\sqrt{n}2^{-t}$ at Equation 31 in the proof of theorem 32. Hence, there exists a choice of $t = O(\log(n))$ so that the discrepancy of the above sequence is at most n^{-4} , thereby satisfying Equation 31. Hence, the statement of Theorem 32 holds for $\mathcal{E} = \mathcal{E}_t$, for $t = O(\log(n))$.

This implies that the algorithm can be computed in $\tilde{O}(n^{\omega+1})$ arithmetic operation, where each operation is a binary operation between two registers of size $O(\log(n))$. Hence, the boolean complexity of the algorithm is $\tilde{O}(n^{\omega+1}) \cdot \log(n) = \tilde{O}(n^{\omega+1})$. \square

\square

7.1 Supporting Claims

Fact 44. Let A be a δ -separated matrix, w a unit vector satisfying

$$\|Aw - cw\| \leq B, \quad B \leq \delta^2/100,$$

for some $|c| \leq 1$. Then there exists $\lambda \in \mathcal{L}(A)$ such that

$$|c - \lambda| \leq \delta/10,$$

and there exists some unit eigenvector v of λ such that

$$\|v - w\| \leq \delta/30.$$

Proof. Without loss of generality, we assume $w = \alpha_1 v_1 + \alpha_2 v_2$ where v_1, v_2 are two distinct elements of some orthonormal basis of A , corresponding to eigenvalues λ_1, λ_2 . Then

$$\|Aw - cw\|^2 = \|v_1 \alpha_1 (c - \lambda_1) + v_2 \alpha_2 (c - \lambda_2)\|^2 \leq B^2$$

By the fact that v_1, v_2 are orthogonal:

$$|\alpha_1|^2 \cdot |c - \lambda_1|^2 + |\alpha_2|^2 \cdot |c - \lambda_2|^2 \leq B^2. \quad (37)$$

Since $\|w\| = 1$ then $|\alpha_1|^2 + |\alpha_2|^2 = 1$, so for at least one $i \in \{1, 2\}$ we have

$$|c - \lambda_i|^2 \leq 2B^2.$$

Suppose w.l.o.g. that $i = 1$. Since A is δ separated then $|\lambda_1 - \lambda_2| \geq \delta$. Then together with the triangle inequality we have

$$|\lambda_2 - c| \geq |\lambda_2 - \lambda_1| - |c - \lambda_1| \geq \delta - \sqrt{2}B \geq 0.9\delta.$$

Hence by Equation 37 above:

$$|\alpha_2|^2 \leq B^2/(0.9\delta)^2 \leq \delta^2/1000$$

This implies that for some unit eigenvector v of λ_1 we have

$$\|w - v\| \leq \delta/30,$$

Using this inequality back in the assumption, and since $\|A\| \leq 1, |c| \leq 1$ then

$$\|cw - Aw\| = \|c(v + \mathcal{E}) - A \cdot (v + \mathcal{E})\| \leq B, \quad \|\mathcal{E}\| \leq \delta/30$$

and since $Av = \lambda_1 v$ then

$$\|cv - \lambda_1 v\| \leq B + 2\delta/30,$$

and since $\|v\| = 1$ we get:

$$|c - \lambda_1| \leq B + 2\delta/30 \leq \delta/10.$$

□

8 Numerical Experiments

In the work above we have outlined a new theoretical approach for solving the eigen-problem of Hermitian matrices. Given the immense practical importance of this problem, we now provide in addition some numerical evidence about the performance of the algorithm. We run the main algorithm for *ASD* of a random matrix A for $n = 20$, where the number of bits of precision used is standard Matlab single precision of 32 bits, and require output precision of $\delta = 10^{-4}$. We then run $50 \cdot n \cdot \log n \sim 2500$ copies of the filter procedure $Filter(A, m, \delta)$. Then, we iterate over a sequence of values for the integer M , i.e. the number such that such that by Theorem 32 a uniformly random integer $m \sim U[M]$ separates any eigenvalue of a perturbed version of A w.h.p.

One can see that while the analytic behavior requires M to be at least n^{75} , here we see that even $M = n^5$ already suffices for the algorithm to return an approximate eigenvector for each eigenvalue w.p. very close to 1. We conjecture, hence, that implementing the algorithm will result in far better practical run time than the analytical bounds we provide here.

Acknowledgements

The authors thank Naomi Kirshner and Robin Kothari for very helpful comments, and Yosi Atia for pointing to us an error in an earlier version of this paper. We also thank anonymous reviewers for their helpful comments and suggestions. This research project was supported in part by the Israeli Centers of Research Excellence (I-CORE) program (Center No. 4/11), by the Israeli Science Foundation (ISF) research grant 1446/09, by an EU FP7 ERC grant (no.280157), and by the EU FP7-ICT project QALGO (FET-Proactive Scheme). LE is thankful to the Templeton Foundation for their support of this work.

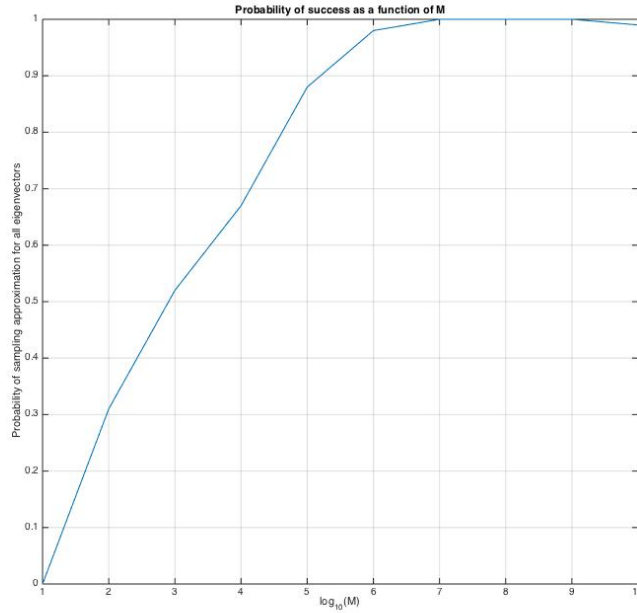


Figure 1: Algorithm Behavior

References

- [Akl90] Selim G. Akl. *Parallel Sorting Algorithms*. Academic Press, Inc., Orlando, FL, USA, 1990.
- [BP92] D. Bini and V. Pan. Practical improvement of the divide-and-conquer eigenvalue algorithms. *Computing*, 48(1):109–123, 1992.
- [DDH07] James Demmel, Ioana Dumitriu, and Olga Holtz. Fast linear algebra is stable. *Numerische Mathematik*, 108(1):59–91, 2007.
- [GVL96] Gene H. Golub and Charles F. Van Loan. *Matrix Computations (3rd Ed.)*. Johns Hopkins University Press, Baltimore, MD, USA, 1996.
- [HHL09] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.*, 103:150502, Oct 2009.
- [Koz92] Dexter C. Kozen. *The Design and Analysis of Algorithms*. Springer-Verlag New York, Inc., New York, NY, USA, 1992.
- [LMM99] Mauro Leoncini, Giovanni Manzini, and Luciano Margara. Parallel complexity of numerically accurate linear system solvers. *SIAM Journal on Computing*, 28(6):2030–2058, 1999.
- [Nie92] H. Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods*. Society for Industrial and Applied Mathematics, 1992.
- [NTV16] Hoi Nguyen, Terence Tao, and Van Vu. Random matrices: tail bounds for gaps between eigenvalues. *Probability Theory and Related Fields*, pages 1–40, 2016.
- [Rei05] John H. Reif. Efficient parallel factorization and solution of structured and unstructured linear systems. *Journal of Computer and System Sciences*, 71(1):86 – 143, 2005.

- [SS90] Gilbert W. Stewart and Jiguang Sun. *Matrix perturbation theory*. Computer science and scientific computing. Academic Press, Boston, 1990.
- [TB97] Lloyd N. Trefethen and David Bau. *Numerical linear algebra*. Society for Industrial and Applied Mathematics, Philadelphia, 1997.
- [TS13] Amnon Ta-Shma. Inverting well conditioned matrices in quantum logspace. In *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing, STOC '13*, pages 881–890, New York, NY, USA, 2013. ACM.
- [Wil12] Virginia Vassilevska Williams. Multiplying matrices faster than coppersmith-winograd. In *Proceedings of the Forty-fourth Annual ACM Symposium on Theory of Computing, STOC '12*, pages 887–898, New York, NY, USA, 2012. ACM.